

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED	
	April 14, 1997	Final Technical, 6/1/93 - 12/31/96	
4. TITLE AND SUBTITLE		5. FUNDING NUMBERS	
Intelligent Access to Scalable Cooperative Information Systems		G N00014-93-1-0863	
6. AUTHOR(S)			
Wesley W. Chu			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)		8. PERFORMING ORGANIZATION REPORT NUMBER	
University of California, Los Angeles Department of Computer Science 405 Hilgard Avenue Los Angeles, CA 90095		940258-00-A02	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
Department of the Navy Office of Naval Research San Diego Regional Office, Suite 300 San Diego, CA 92121-3019			
11. SUPPLEMENTARY NOTES		19970425 035	
12a. DISTRIBUTION/AVAILABILITY STATEMENT		12b. DISTRIBUTION CODE	
Approved for public release. Distribution is unlimited.		DTIC QUALITY INSPECTED 2	
13. ABSTRACT (Maximum 200 words) The AASERT grant, "Intelligent Access to Scalable Cooperative Information Systems" enabled the UCLA graduate student Michael Minock to address his dissertation toward the role of Explanation and Paraphrase technology in Cooperative Information Systems. Given the requirement of providing explanation for CoBase[2] answers, this work furthered the overall mission of our parent DARPA contract. In the first year of the AASERT we built a prototype of CoBase capable of generating explanations and descriptions which we demonstrated at the 1994 DARPA Planning Workshop in Tucson. In the second year the technology in the prototype was extended and transferred to a modular explanation server and integrated with the full CoBase system. The extensibility this system was established by providing explanation for CoBase over Electronic Warfare and Medical Image domains[4]. Currently the Explanation system is undergoing rigorous testing by other graduate students in our group. Theoretical work focused on the representation of CoBase's trace and the process of computing explanations and generating descriptions[5]. We have produced an instance classification system over which explanation is provably complete, correct, and minimal[6]. Progress has also been made toward an Object-Oriented framework[7] that enables complete and correct description and explanation for applications such as CoBase.			
14. SUBJECT TERMS		15. NUMBER OF PAGES 6	
Explanation system, Cooperative Database System, Intelligent Information System.		16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT unclassified		18. SECURITY CLASSIFICATION OF THIS PAGE unclassified	
		19. SECURITY CLASSIFICATION OF ABSTRACT unclassified	
		20. LIMITATION OF ABSTRACT	

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to stay within the lines to meet optical scanning requirements.

Block 1. Agency Use Only (Leave blank).

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C - Contract	PR - Project
G - Grant	TA - Task
PE - Program Element	WU - Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. (If known)

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement.

Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."

DOE - See authorities.

NASA - See Handbook NHB 2200.2.

NTIS - Leave blank.

Block 12b. Distribution Code.

DOD - Leave blank.

DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.

NASA - Leave blank.

NTIS - Leave blank.

Block 13. Abstract. Include a brief (Maximum 200 words) factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code (NTIS only).

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

Explanation and Paraphrase Technology for Cooperative Information Systems

Wesley Chu

April 14, 1997

Abstract

The AASERT grant, "Intelligent Access to Scalable Cooperative Informations Systems" enabled the UCLA graduate student Michael Minock to address his dissertation toward the role of *Explanation* and *Paraphrase* technology in Cooperative Information Systems. Given the requirement of providing explanation for CoBase[2] answers, this work furthered the overall mission of our parent DARPA contract.

In the first year of the AASERT we built a prototype of CoBase capable of generating explanations and descriptions which we demonstrated at the 1994 DARPA Planning Workshop in Tuscon. In the second year the technology in the prototype was extended and transferred to a modular explanation server and integrated with the full CoBase system. The extensibility this system was established by providing explanation for CoBase over Electronic Warfare and Medical Image domains[4]. Currently the Explanation system is undergoing rigorous testing by other graduate students in our group.

Theoretical work focused on the *representation* of CoBase's trace and the *process* of computing explanations and generating descriptions[5]. We have produced an instance classification system over which explanation is provably complete, correct, and minimal[6]. Progress has also been made toward an Object-Oriented framework[7] that enables complete and correct description and explanation for applications such as CoBase.

1 Introduction

CoBase[2] provides cooperative (summary, approximate, or relaxed) answers to queries that may contain cooperative operations (e.g. similar-to, near-to) and conceptual terms. Data are organized into conceptual (type abstraction) hierarchies based on the database schema, application characteristics

and user type. The higher levels of the hierarchy provide a more abstract representation of data than the lower levels. Generalization (moving up in the hierarchy), specialization (moving down in the hierarchy) and association (moving between hierarchies) are the three key operations in deriving cooperative query answers.

Relaxation in CoBase can be specified explicitly by the user or calling program, or by query modification via generalization, specialization or association. The cooperative operators are available as extensions to several existing query languages, including CoSQL (cooperative SQL) and CoOQL (cooperative object query language). If the approximate operations are not explicitly specified, or if no exact answer to the query is found, CoBase modifies queries via generalization, specialization or association based on the type abstraction hierarchies. CoBase may use default knowledge about the user and context to control this process. CoBase supports relational databases (e.g. Oracle and Sybase). We have demonstrated the feasibility and functionality of CoBase on top of a Transportation Planning Database and in the domain of logistics planning. CoBase technology has successfully been integrated into the GLAD system and was demonstrated in the IFD 1.2 July 1996.

Similar to a human expert, CoBase should be able to offer explanations of the answers it yields. If a user submits a high-level or imprecise query (e.g. similar-to or near to operators) and is presented with the set of answers alone, the user might ask - "why these answers?" Such a question demands an explanation. Explanation Systems[9] provide dynamic information as opposed to the static information of traditional *help* systems. This includes explanations and descriptions of how the system interpreted the user's question and how the search for answers was conducted. In the case that the user's question is ill-posed, the system should be able to offer an explanation of why. In the case that an answer did not exist, the quality of approximate answers needs to be explained. Alternatively if there are too many answers, then the presentation of a subset of answers will need to be explained as being particularly suited to the user. We use a transportation database to illustrate the problem. Such a database includes information about planes, ships, airports, seaports, cargo characteristics, etc.

Suppose the user poses the query: "*Is there an airport in Gafsa, Tunisia with a runway length greater than 8500'?*"

It happens that there is no such airports, so CoBase relaxes the location name attribute to find nearby airports with runway length greater than

8500'. The following is presented.

S1: *There is no airport where location is Gafsa and runway length is greater than 8500 feet.*

S2: *There is no airport where region is South West Tunisia and runway length is greater than 8500 feet.*

S3: *Through the relaxation of Gafsa to all location names in South Tunisia an airport in location Gabes with runway length equal to 9126 feet has been found.*

S4: *The whole region of Tunisia could still be searched for an airport where runway length is greater than 8500 feet.*

The user may select text fragments to obtain further explanation and description. Although not shown here, sensitive text fragments are stacked hierarchically, letting the user interact with a single word, phrase, sentence or paragraph. When the user ‘clicks’ the text “Through the relaxation of Gafsa to South Tunisia” the following options are presented.

- 1.) *Show map of South Tunisia including Gafsa and Gabes*
- 2.) *Give more detailed description of relaxation action*

If the first option is clicked then a map appears, and if the second option is clicked then a more specific description of exactly how the query was relaxed is provided.

2 Initial Proof of Concept

With the motivation of integrating Explanation and Cooperative Information systems established, in the first year of the AASERT proposal we focused on building a prototype of CoBase capable of generating explanations and descriptions. This prototype was implemented in LISP and LOOM[3] and used the ISI Aspect Text planner[8] for natural language generation. The LOOM classifier was used as an inference engine that formed a basis of explanation through classification. The user interface was written in CLIM and the SIMS[1] system handled the underlying relational queries dispatched by CoBase.

This prototype was demonstrated at the DARPA Planning Workshop held in Tuscon Arizona in February of 1994. The system was well received and performed robustly for queries requiring limited forms of relaxation.

3 Engineering and Formalism

Following the initial proof of concept, the research work split to address *practical* aspects of bringing explanation to CoBase as well as formal, *theoretical* issues.

On practical grounds the proof of concept prototype was limited. It stood as an independent system that had only been integrated with a scaled down CoBase, not the full CoBase system of our parent grant. The system was written in LISP and LOOM and was heavy weight and slow. To remedy these short-comings the technology embodied in the initial proof of concept was transferred to a modular explanation server written in C++ and CLIPS. This server was then integrated with a new, more stable version of CoBase being written in C++. A model of query processing was built specifically to interpret CoBase operations and trace-generating operations were added to existing CoBase routines. This modularized the explanation system and reduced its impact on the design and implementation of other CoBase components. The performance of the new explanation system was also superior to the initial proof of concept system. Finally the extensibility of CoBase's explanation system was exhibited when other graduate students applied CoBase to Medical Image and Electronic Warfare domains and received useful explanations.

The formal, theoretical development focused on the *representation* of the trace of CoBase objects and actions and the *process* of computing explanations and of generating linguistically sufficient descriptions from this trace. Deeper theoretical issues in explanation and natural language generation were also explored[4]. In this phase of the research an extensive literature review was also conducted. The ongoing formal work has been integrated into the working system. In addition to enabling deeper more complete explanation, this work has also disciplined and clarified the implementation, thus easing maintenance and integration costs.

This period spanned the second and into the third year of the AASERT contract. This work is documented in a paper[5].

4 Synthesis

The last year of the AASERT grant has produced more results. On the theoretical side, we have developed a well defined instance classification system over which explanation is provably complete, correct, and minimal[6]. Elements of this theory are being incorporated into CoBase's explanation system. Another direction is an Object-Oriented framework that enables complete description and explanation[7]. This framework gives applications, such as CoBase, a set of design and implementation requirements that, if adhered to, enable automatic description and explanation of system inputs, outputs, actions, and mechanism.

Finally the integration of explanation and description with the whole CoBase system is nearing completion and is undergoing rigorous testing by other graduate students in our group. A major focus of this work is guaranteeing completeness and correctness of explanation over the actual CoBase system of our parent contract.

5 Conclusions

The AASERT grant, "Intelligent Access to Scalable Cooperative Information Systems" enabled Michael Minock to address his dissertation toward the role of *Explanation* and *Paraphrase* technology in Cooperative Information Systems. This lead to a system that provides the CoBase System with robust explanation and description capabilities. This work also makes theoretical and practical contributions toward provably correct and complete explanation.

The work presented here is relevant to the CoBase project and would not have been carried out but for the AASERT program. AASERT is a particularly effective program for insuring the production of U.S. Ph.D.'s in Computer Science. Topics are motivated by, and directed toward, providing useful results for the parent grant. Furthermore, in an era of increased university fees and hidden expenses, AASERT's *complete* coverage of student expenses reduces the probability of capable students being enticed into entering industry prematurely.

References

- [1] Y. Arens and C. Knoblock. Planning and reformulating queries for semantically-modelled multidatabase systems. In *Proceedings*. First

International Conference on Information and Knowledge Management (CIKM), 1992.

- [2] W. Chu, H. Yang, Chiang K., M. Minock, G. Chow, and C. Larson. Cobase: A scalable and extensible cooperative information system. *Intelligent Information Systems*, 6(3):223–259, 1996.
- [3] R. MacGregor and M. Burstein. Using a descriptive classifier to enhance knowledge representation. *IEEE Expert*, 6(3):41–47, 1991.
- [4] M. Minock and W. Chu. Generation, refinement, and extension of explanations for cooperative information systems. In *Proceedings. CIKM Workshop on New Paradigms in Information Visualization and Manipulation*, 1995.
- [5] M. Minock and W. Chu. Interactive explanation for cooperative information systems. In *Lecture Notes in AI. Proceedings of the 9th International Symposium on Methodologies for Intelligent Information Systems*, June 1996.
- [6] M. Minock and W. Chu. Computing conditional explanations over a classification hierarchy. CoBase Technical Note, April 1997.
- [7] M. Minock and W. Chu. An object-oriented framework supporting correct and complete explanation and description. CoBase Technical Note, April 1997.
- [8] J. Moore and C. Paris. Planning text for advisory dialogue. In *Proceedings of the Twenty-Seventh Annual Meeting of the Association for Computational Linguistics*, British Colombia, 1989.
- [9] W. Swartout, C. Paris, and J. Moore. Design for explainable expert systems. *IEEE Expert*, 6(3):58–64, 1991.